

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of
Dent

Serial No.: **09/727,062**

Filed: **November 30, 2000**

For: **ANTI-SPOOFING PASSWORD
PROTECTION**

Attorney's Docket No: **4015-721**

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

)
) Patent Pending
)
) Examiner: Peter Poltorak
)
) Group Art Unit: 2134
)
) Confirmation No.: 2720
)
)
)

CERTIFICATE OF MAILING OR TRANSMISSION [37 CFR 1.8(a)]

I hereby certify that this correspondence is being:

☐ deposited with the United States Postal Service on the date shown below with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

☐ transmitted by facsimile on the date shown below to the United States Patent and Trademark Office at (703) 273-8300.

October 11, 2006

Date

Kathleen Koppen
Kathleen Koppen

This correspondence is being:

☒ electronically submitted via EFS-Web

APPEAL BRIEF

Sir:

This Appeal Brief is being submitted not more than two months after the Patent Office received the Notice of Appeal (August 11, 2006). As such, no extension of time fees should be due. The requisite fee for filing this Brief is being submitted concurrently with this paper. If the submitted fee is insufficient, the Commissioner is authorized to charge the difference to Deposit Account 18-1167.

(1) REAL PARTY IN INTEREST

The real party in interest is Ericsson, Inc., the assignee of the present invention.

(2) RELATED APPEALS AND INTERFERENCES

There are no related appeals or interferences to the best of Applicants' knowledge.

(3) STATUS OF CLAIMS

A total of twenty-one (21) claims numbered 1-21 have been presented for examination. Claims 1-19 were originally filed with the present application. During prosecution, Applicant added claims 20-21, and cancelled claims 6, 13, 16, 19, and 21. Currently, claims 1-5, 7-12, 14-15, 17-18, and 20 stand finally rejected by the Examiner. Accordingly, Applicants appeal the final rejection of claims 1-5, 7-12, 14-15, 17-18, and 20.

(4) STATUS OF AMENDMENTS

All amendments have been entered to the best of Applicant's knowledge.

(5) SUMMARY OF CLAIMED SUBJECT MATTER

The claimed invention relates to a system and method of protecting passwords from inadvertent or unintentional disclosure to a foreign party. *Spec.*, p.1, ll. 1-3; p.2, ll. 16-17. A security module in a computing device capable of performing a password-protected secure function, for example, may implement the claimed method. *Spec.*, p.3, ln. 15 – p.6, ln. 2; Figure 1; *see also Spec.*, p. 5, ln. 15 – p. 9, ln. 10; Figure 2.

In one embodiment, the security module comprises a secure processor (112), memory (114, 116), and a display (20). The memory and the display are communicatively coupled to the secure processor. *Spec.*, p.4, ln. 17 – p.5, ln. 4 and Figure 1; p.6, ll. 9-13 and Figure 2. The secure memory, which may comprise a flash EEPROM and/or a Read Only Memory (ROM), for example, stores authentication indicia for authenticating password entry screens (150) to the user of the device. *Spec.*, p. 7. ln. 18 – p. 8, ln. 6; Figure 3. The authentication indicia functions

as a “reverse password” that is used by the secure processor to authenticate valid password entry screens. Particularly, the presence or absence of the authentication indicia would alert a user as to whether a password entry screen on the display is legitimate. Because the authentication indicia are stored in secure memory, it is not possible for a rogue program to spoof the password entry screen. *Spec.* p. 10, ll. 13-24; Figure 3.

In operation, the secure processor receives a user command (e.g., a service request) to execute a password-protected secure function. The secure processor then executes a password program stored in the ROM or flash EEPROM to obtain a password associated with the password-protected secure function. The secure processor also outputs a password entry screen that contains authentication indicia retrieved from the flash EEPROM. To prevent fraudulent parties from “spoofing” a password entry screen, the secure processor temporarily halts the execution of any programs not needed by the secure processor while the password entry screen is displayed. Once the user enters a password, the secure processor checks its validity. If valid, the secure processor executes the password protected secure function for the user, and removes the password entry screen from the display. The secure processor then restarts any halted programs after the password entry screen is removed from the display. *Spec.*, p. 10, ll. 13-19; p. 13, ln. 14 – p. 14, ln. 23; p. 15, ln. 1 – p. 16, ln. 19; Figures 3, 5.

A variety of techniques may be used to temporarily halt application programs not needed by the secure processor while the password entry screen is displayed. For example, a security lock program may inhibit all processor interrupts, except for keyboard and display interrupts responding to a request by the security module. Alternatively, the security lock program could prevent context-switching by the operating system, or manipulate the settings in a status table used by the operating system. In one embodiment, the claimed invention directs the operating system to use an alternative status table while the password entry screen is displayed. *Spec.*, p. 15, ln. 1 – p. 16, ln. 19.

(6) GROUNDS OF REJECTION

The Examiner finally rejected claims 1-5, 9-11, and 17-18 under 35 U.S.C. §103(a) as obvious over Windows NT as evidenced by the article entitled “Windows NT Server 4 Security Handbook” authored by Hadfield (hereinafter “Hadfield”) and the web site page entitled “NT Workstation Resource Kit” (<http://web.archive.org/web/20000831044112/is-it-true.org/nt/atips/atips71.shtml>) (hereinafter, “NT Resource Kit”), in view of U.S. Pat. No. 5,652,890 to Foster (hereinafter, “Foster”).

The Examiner finally rejected claims 7-8 and 14-15 under 35 U.S.C. §103(a) as obvious over Windows NT as evidenced by Hadfield and NT Resource Kit in view of Foster, and in further view of the article to Pfleeger entitled “Security in Computing” (hereinafter, “Pfleeger”).

(7) ARGUMENTS RELATING TO THE §103(a) GROUND OF REJECTION

A. Hadfield, the NT Resource Kit, and Foster fail to teach or suggest, alone or in combination, the authentication indicia of claim 1.

Claim 1 is directed to a method of performing a password-protected secure function. The method may be implemented by a security module in a computing device. For reference, claim 1 appears below.

1. A method implemented by a security module in a computing device of performing a password-protected secure function, said method comprising:
 - storing authentication indicia for authenticating password entry screens to a user in a memory of the computing device;
 - receiving a command to execute a password-protected secure function;
 - temporarily halting execution of programs not needed by the security module while the data entry screen is displayed;
 - prompting the user to enter a password associated with the secure function by displaying a password entry screen containing the authentication indicia responsive to receiving the command;
 - removing the data entry screen from the display;
 - restarting halted programs after the password entry screen is removed from the display; and
 - executing the password-protected secure function based on the validity of the password entered by the user.

The authentication function of claim 1 is performed by including the authentication indicia on the password entry screen when the password entry screen is displayed. The user can differentiate a valid password entry screen from an invalid or “spoofed” password entry screen by the presence of the authentication indicia. More specifically, a valid password entry screen would always contain the authentication indicia while an invalid password entry screen would not.

The Examiner equates the claimed authentication indicia to a user name on a Windows NT password entry screen. The Examiner’s contention is that the presence or absence of the user name authenticates the Windows NT password entry screen as valid or invalid to the user. This contention is wholly incorrect and completely unsupported by the cited art.

The Hadfield reference is a handbook that details security system operation on Windows NT. The portions of this handbook provided to the Applicant specifically relate to a well-known login process in which a user presses a Control-Alt-Delete sequence of keys on a keyboard to invoke the initial login screen. *Hadfield*, p. 81, ¶¶1-2.

Hadfield does not teach or suggest that the user name is authentication indicia, let alone the claimed authentication indicia. The user name on the NT password entry screen of Hadfield – or any NT password entry screen for that matter - is used only to authenticate the user to the computer. The presence or absence of a user name on an NT password entry screen does nothing to authenticate that screen as valid to the user. In fact, the presence of a user name on an NT password entry screen actually constitutes a security threat. The other reference relied on by the Examiner to show Windows NT functionality - NT Resource Kit - expressly evidences this fact. “By default, Windows NT 4.0 displays the name of the last person who logged on to the system. This informational exposure can pose a security threat, especially if a user's password can be guessed from the account name or the login environment.” *NT Resource Kit*, ¶1 (emphasis added).

The NT Resource Kit further evidences the conclusory nature of the Examiner's assertion by explicitly providing the name of a function "DontDisplayLastUserName" that disables Windows NT from displaying a user name. *NT Resource Kit*, ¶12. That is, rather than displaying a user name to authenticate the password entry screen as the Examiner asserts, the NT Resource Kit reference actually teaches removing the user name from the password entry screen to address the security issues. .

Thus, the Examiner's assertion that a user name on the Windows NT password entry screen is the claimed authentication indicia is wholly unsubstantiated. Indeed, Hadfield never suggests that a user name is authentication indicia, and the NT Resource Kit explicitly contradicts that assertion. User names and passwords are not secure, and therefore, user names are just as prone to being spoofed as the password entry screen itself. As evidenced by the very art the Examiner uses to support the §103 rejection of claim 1, a user has no way of knowing whether a conventional NT password entry screen is valid or invalid based merely on the presence or absence of a user name. Any contention to the contrary is mere conjecture, and as the Board is well aware, mere conjecture can never support a §103 rejection.

The final cited reference, Foster, does nothing to remedy these deficiencies of Windows NT or the other cited art. Foster discloses generating an interrupt to force a microprocessor in a laptop computer to switch in and out of a protected mode. Foster does not teach or suggest including authentication indicia in a password entry screen to authenticate that screen as valid to the user, and the Examiner never asserts that it does.

Therefore, none of the cited references teaches or suggests, alone or in combination, the claimed authentication indicia. As such, none of the cited references teaches or suggests, alone or in combination, a password entry screen containing authentication indicia to validate the authenticity of that screen to a user. Accordingly, the §103 rejection of claim 1 should be withdrawn.

B. Hadfield, the NT Resource Kit, and Foster fail to teach or suggest, alone or in combination, temporarily halting the execution of programs not needed by a security module while a password entry screen is displayed claim 1.

There is another reason why the §103 rejection of claim 1 should be withdrawn. Particularly, claim 1 requires that temporarily halting the execution of programs not needed by a security module while a password entry screen is displayed. The Examiner admits that neither Hadfield nor the NT Resource Kit teaches or suggests, alone or in combination, this aspect of claim 1, but alleges that Foster does. Foster does not.

According to Foster, computers not running in a protected mode may suspend an application program when power to the computer is turned off. A microprocessor saves the state of the suspended application program in memory such that when power is restored, the microprocessor may resume executing the application program from the point at which it was suspended. However, Foster recognizes that microprocessors running in a protected mode are restricted from accessing some portions of memory, and thus, may not be able to save the state information when the power is turned off. Therefore, Foster generates an interrupt to implement the suspend/resume capabilities for a microprocessor running in a protected mode. *Foster*, col. 9, ln. 22 – col. 10, ln. 8.

Foster generates the disclosed interrupts to force a microprocessor in and out of a protected mode responsive to a power disruption. This allows a microprocessor running in a protected mode to access the requisite portions of memory to effect a suspend/resume operation. However, whatever Foster discloses with respect to suspend/resume operations, one thing is certain. The user has already successfully logged on to the computer. That is, at the point that Foster teaches generating the interrupts, the user has already navigated beyond the password entry screen and is using application programs.

It appears that the Examiner cites Foster only to support an allegation that temporarily halting the execution of programs not needed by a security module while a password entry

screen is displayed is known. In light of the Foster reference, however, allegation fails scrutiny.

Foster does not mention anything about temporarily halting the execution of programs not needed by a security module while a password entry screen is displayed. Indeed, whatever Foster does teach occurs after the password entry screen has been removed from the display.

Therefore, none of the cited references teaches or suggests, alone or in combination, temporarily halting the execution of programs not needed by a security module while a password entry screen is displayed. Accordingly, for this additional reason, the §103 rejection of claim 1 should be withdrawn.

C. There is no motivation to combine Hadfield, the NT Resource Kit, and Foster.

The Examiner asserts that one skilled in the art would be motivated temporarily halt the execution of programs not needed by the security module “to prevent others from examining information (used in the authentication process) without exiting the currently [running] application programs.” *Final Office Action*, p. 6, ll. 7-11. However, this motivation fails scrutiny.

None of the references by themselves teaches or suggests displaying a password entry screen containing authentication indicia or temporarily halting the execution of programs not needed by a security module while a password entry screen is displayed. Therefore, combining them cannot yield each and every limitation of claim 1.

Moreover, modifying the Hadfield and NT Resource Kit references as the Examiner asserts still fails to produce the method of claim 1. Claim 1 temporarily halts running programs while the password entry screen is displayed to prevent malicious attackers from spoofing the password entry screen. Contrastingly, any Foster-type interrupts generated by the Examiner’s “modified” product would not occur until after the user is logged on to the computer, and after the password entry screen has been removed from the display. Therefore, even with the Examiner’s alleged combination, a malicious attacker could have already “spoofed” the password entry screen to obtain the user’s password without the user’s knowledge. Indeed,

assuming that the references could be combined at all, there is no real chance that the proffered combination would be successful at protecting passwords from inadvertent or unintentional disclosure to a foreign party.

Simply put, there is no motivation to combine the references. Accordingly, the §103 rejection of claim 1 should be withdrawn for this additional reason.

D. Hadfield, the NT Resource Kit, and Foster do not render claim 11 obvious.

Claim 11 is directed to a device that executes a password-protected secure function. The device comprises a secure processor, memory configured to store authentication indicia, and a display to display a password entry screen containing the authentication indicia to the user. For reference, claim 1 appears below.

11. A device for executing a password-protected secure function comprising:
 - a secure processor configured to receive a command to execute a password-protected secure function, and to execute a password program to obtain a password associated with the password-protected secure function from a user responsive to receiving the command;
 - memory operatively connected to the secure processor and configured to store authentication indicia for authenticating password entry screens to the user of the device;
 - a display operatively connected to the secure processor; and
 - the secure processor configured to:
 - output a data entry screen containing said authentication indicia to said display;
 - temporarily halt execution of programs not needed by the secure processor while the password entry screen is displayed;
 - remove the data entry screen from the display;
 - restart halted programs after the password entry screen is removed from the display; and
 - execute the password-protected secure function based on the validity of the password entered by the user.

In claim 11, the secure processor outputs a password entry screen containing the authentication indicia. To promote security, the secure processor temporarily halts the execution of programs not needed by the secure processor. It does not restart those programs until after the password entry screen is removed from the display.

The Examiner rejected claim 11 as being obvious over Windows NT as evidenced by Hadfield and NT Resource Kit in view of Foster. However, for reasons similar to those stated above with respect to claim 1, none of the references teaches or suggests, alone or in combination, claim 11. Accordingly, the §103 rejection of claim 11 should also be withdrawn.

E. The Examiner's articulation of the §103 rejection of claim 11 is legally insufficient.

Additionally, the §103 rejection of claim 11 also fails for another reason. Specifically, the §103 rejection to claim 11 is *legally insufficient*. In rejecting claim 11, the Examiner states only that, "As per claim 11 the processor working in the authentication mode reads on a secure processor." *Final Office Action*, p. 6, ll. 16-17. That single, naked allegation constitutes the full extent of the rejection to claim 11. There are no citations to the references or any other evidence of record, and the Examiner never even attempts to provide a motivation to combine the references.

The law regarding §103 rejections is well-settled. To establish a *prima facie* case of obviousness, three basic criteria must be satisfied. First, the cited reference(s) must teach or suggest each and every limitation of a claimed invention. Second, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to those skilled in the art, to modify the reference(s). In advancing a motivation to modify the cited reference(s), the Examiner must provide some suggestion of the desirability of modifying the reference(s). This suggestion cannot come from Applicants' own disclosure. Finally, there must be some reasonable expectation of success (*see MPEP* §706.02(j)).

Regarding a motivation to combine references, the Examiner must provide at a minimum, "an explanation based on logic and sound scientific reasoning that will support a holding of obviousness." *Ex parte Levengood*, 28 USPQ2d 1300, 1301 (Bd. Pat. App. & Inter. 1993) (emphasis added). The conclusory statement provided by the Examiner regarding a

“processor working in the authentication mode” is woefully inadequate with respect to addressing a §103 rejection, and utterly fails to fulfill the legal requirements set forth by law. Therefore, for this additional reason, the §103 rejection of claim 11 fails as a matter of law and should be withdrawn.

F. Hadfield, the NT Resource Kit, and Foster fail to teach or suggest, alone or in combination, storing authentication indicia in a security module as in claim 2.

Claim 2 further requires that the authentication indicia of claim 1 be stored in a security module. The claimed security module is a secure device, such as a tamper-proof chip, for example, that performs security functions. The security module may comprise a removable smart card that inserts into or connects with a removable data storage device or other interface. The security functions performed by security module include, but are not limited to, one or more of the following services: encryption and decryption of data, authentication of user identities, key generation and management, password authentication, and data integrity verification. For reference, claim 2 appears below.

2. The method of claim 1 wherein storing authentication indicia recognized by said user in said computing device comprises storing said authentication indicia in a security module.

Claim 2 is dependent from claim 1, which as stated above, is patentably non-obvious over the cited art. As such, the §103 rejection of claim 2 fails as a matter of law.

Nevertheless, the Examiner asserts that, according to Hadfield, Windows NT stores “authentication indicia” (i.e., user names and passwords) in a user accounts database (Security Accounts Manager - SAM). The Examiner also asserts that Windows NT includes a security module that stores authentication indicia obtained from a user. *Final Office Action*, p. 5, ¶18.

Whatever Hadfield teaches with respect to Windows NT security, it does not teach or suggest claim 2. The SAM database is simply a user account database that may store user

names and passwords. As stated above, user names are not the claimed authentication indicia.

Moreover, the Examiner's characterization of the Windows NT Security Subsystem is misleading. The subsystem is a simply collection of software modules. It is not a security module as the specification describes that term. *See Spec.*, p. 5, ln. 15 – p. 9, ln. 10; Figure 2.

Hadfield does not teach or suggest the claimed security module. Nor do the other cited references, and the Examiner does not assert that they do. Therefore, the §103 rejection of claim 2 should be withdrawn.

G. Hadfield, the NT Resource Kit, and Foster fail to teach or suggest, alone or in combination, claims 9 and 17.

Claims 9 and 17 depend directly from claims 1 and 11, respectively, and further define the temporarily halting step. For reference, claims 9 and 17 appear below in their entirety.

9. The method of claim 1 wherein temporarily halting execution of programs not needed by said security module while said password entry screen is displayed comprises:
- storing a status table in random access memory used by an operating system in said computing device, each entry in said status table relating to a currently executing program and containing a status indication associated with said currently executing program;
 - saving current settings of said status table; and
 - changing said current settings so as to inhibit execution by said operating system of said programs not needed by said security module.

17. The device of claim 11 wherein said secure processor halts execution of programs not needed by said secure processor to obtain said password from said user by changing settings in a status table used by an operating system while said password entry screen is displayed.

The Examiner rejected claims 9 and 17 as being obvious over Windows NT as shown by Hadfield, the NT Resource Kit, and Foster. Interestingly, however, the Examiner admits that none of these references, alone or in combination, teaches or suggests claims 9 and 17. Instead, the Examiner simply takes Official Notice to support the assertion that Operating Systems (OS) are well-known to maintain status tables to track executing processes.

The Examiner has no proof that this particular OS functionality is used when temporarily halting the execution of programs not needed by a security module while a password entry screen is displayed. At most, the only thing the assertion says is that an OS tracks running programs. The Examiner simply alters this alleged functionality without providing any support in an attempt to reject the claims. As the Board is well aware, such conclusionary statements can never rise to the level of being evidence. *In re Dembizek*, 175 F.3d 994, 999, 50 U.S.P.Q. 2d 1614, 1617 (Fed. Cir. 1999). In setting forth a factual basis for motivation to combine, the Patent Office must go beyond mere broad conclusionary statements and set forth specific understandings or technical principals that would motivate a person of ordinary skill in the art to make the combination that would render the combination obvious. *Id.*

With respect to claims 9 and 17, the Examiner has failed to put forth a *legally sufficient* prima facie case of obviousness. Indeed, the §103 rejection of claims 9 and 17 fails as a matter of law.

H. Hadfield, the NT Resource Kit, and Foster fail to teach or suggest, alone or in combination, claims 10 and 18.

Claim 10 depends directly from claim 1 and further defines another embodiment of the temporarily halting step. Claim 18 incorrectly depends from cancelled claim 16, but should depend from claim 11. This minor error will be corrected upon receipt of a Notice of Allowance or another Office Action from the Examiner. For reference, claims 10 and 18 appear below in their entirety.

10. The method of claim 1 wherein temporarily halting execution of programs not needed by said security module while said password entry screen is displayed comprises:

- storing an alternate status table in random access memory used by an operating system in said computing device, each entry in said alternate status table relating to a program needed by said security module;
- instructing said operating system to use said alternate status table while said password entry screen is displayed.

18. The device of claim 16 wherein said secure processor halts execution of programs not needed by said secure processor to obtain said password from said user by causing an operating system to use an alternate status table while said password entry screen is displayed.

Claims 10 and 18 require the use of an alternate status table in random access memory while the password entry screen is displayed. The Examiner rejected claims 10 and 18 as being obvious over Windows NT as shown by Hadfield, the NT Resource Kit, and Foster, but admits that none of those references actually teaches or suggests either claim, alone or in combination. Rather, the Examiner relies on the same Official Notice used to reject claims 9 and 17.

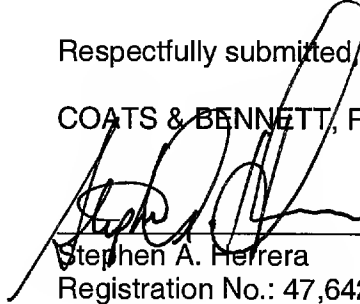
Both the Official Notice and the Examiner's motivation are simply unsupported conjecture. The Examiner has no proof that an OS uses alternate status tables when temporarily halting the execution of programs not needed by a security module while a password entry screen is displayed. Nor does the Examiner have any proof that it would be obvious to do so. Accordingly, the §103 rejections of claims 10 and 18 fails as a matter of law for reasons similar to those stated above. Therefore §103 rejections of claims 10 and 18 must be withdrawn.

Conclusion

For the reasons set forth above, none of the references, alone or in combination, teach or suggest the claimed invention. Accordingly, all claims being appealed herein are patentable over the cited art. Applicant respectfully requests that the Board overturn all rejections.

Respectfully submitted,

COATS & BENNETT, P.L.L.C.



Stephen A. Herrera
Registration No.: 47,642

Dated: October 11, 2006

P.O. Box 5
Raleigh, NC 27602
Telephone: (919) 854-1844
Facsimile: (919) 854-2084

(8) CLAIMS APPENDIX

1. A method implemented by a security module in a computing device of performing a password-protected secure function, said method comprising:

storing authentication indicia for authenticating password entry screens to a user in a memory of the computing device;

receiving a command to execute a password-protected secure function;

temporarily halting execution of programs not needed by the security module while the data entry screen is displayed;

prompting the user to enter a password associated with the secure function by displaying a password entry screen containing the authentication indicia responsive to receiving the command;

removing the data entry screen from the display;

restarting halted programs after the password entry screen is removed from the display;

and

executing the password-protected secure function based on the validity of the password entered by the user.

2. The method of claim 1 wherein storing authentication indicia recognized by said user in said computing device comprises storing said authentication indicia in a security module.

3. The method of claim 1 wherein displaying said password entry screen containing said authentication indicia comprises displaying said authentication indicia for a limited time.

4. The method of claim 1 further comprising obtaining said authentication indicia from said user.

5. The method of claim 1 further comprising halting execution of programs running on said computing device not necessary for inputting said password while said password entry screen is displayed.

6. (Canceled)

7. The method of claim 1 wherein temporarily halting execution of programs not needed by said security module while said password entry screen is displayed comprises inhibiting an operating system in said computing device from responding to interrupts not associated with said security module.

8. The method of claim 1 wherein temporarily halting execution of programs not needed by said security module while said password entry screen is displayed comprises inhibiting context-switching by an operating system in said computing device to programs not needed by said security module.

9. The method of claim 1 wherein temporarily halting execution of programs not needed by said security module while said password entry screen is displayed comprises:

storing a status table in random access memory used by an operating system in said computing device, each entry in said status table relating to a currently executing program and containing a status indication associated with said currently executing program;

saving current settings of said status table; and

changing said current settings so as to inhibit execution by said operating system of said programs not needed by said security module.

10. The method of claim 1 wherein temporarily halting execution of programs not needed by said security module while said password entry screen is displayed comprises:

storing an alternate status table in random access memory used by an operating system in said computing device, each entry in said alternate status table relating to a program needed by said security module;

instructing said operating system to use said alternate status table while said password entry screen is displayed.

11. A device for executing a password-protected secure function comprising:

a secure processor configured to receive a command to execute a password-protected secure function, and to execute a password program to obtain a password associated with the password-protected secure function from a user responsive to receiving the command;

memory operatively connected to the secure processor and configured to store authentication indicia for authenticating password entry screens to the user of the device;

a display operatively connected to the secure processor; and

the secure processor configured to:

output a data entry screen containing said authentication indicia to said display;

temporarily halt execution of programs not needed by the secure processor while the password entry screen is displayed;

remove the data entry screen from the display;

restart halted programs after the password entry screen is removed from the display;

and

execute the password-protected secure function based on the validity of the password entered by the user.

12. The device of claim 11 further comprising a smart card containing said secure processor and said memory.

13. (Canceled)

14. The device of claim 11 wherein said secure processor halts execution of programs by inhibiting an operating system from responding to interrupts not associated with said secure processor while said password entry screen is displayed.

15. The device of claim 11 wherein said secure processor halts execution of programs by inhibiting an operating system from context-switching while said password entry screen is displayed.

16. (Canceled)

17. The device of claim 11 wherein said secure processor halts execution of programs not needed by said secure processor to obtain said password from said user by changing settings in a status table used by an operating system while said password entry screen is displayed.

18. The device of claim 16 wherein said secure processor halts execution of programs not needed by said secure processor to obtain said password from said user by causing an operating system to use an alternate status table while said password entry screen is displayed.

19. (Canceled)

20. The device of claim 11 wherein said secure processor and said memory are contained within a removable security module.

21. (Canceled)

(9) EVIDENCE APPENDIX

There is no further evidence not contained in the prosecution history.

(10) RELATED PROCEEDINGS APPENDIX

There are no related proceedings.